HIGH EFFICIENT CLASSIFICATION ON REMOTE SENSING IMAGES BASED ON SVM

Qiu Zhen Ge [a, b] *, Zhang Chun Ling [e] ,Li. Qiong [c], Xin Xian Hui [d] , Guo Zhang [f]

[a] Key Laboratory of Geo-informatics of State Bureau of Surveying and Mapping, Chinese Academy of surveying and mapping, Beijing, China, 100039
[b] Institute of Computing Technology Chinese Academy of Sciences, Beijing, China, 100080, - qiuzhenge@sina.com
[c] China Institute of Geotechnical Investigation and Surveying, Beijing 100007, China - liq@cigis.com.cn
[d] Tian Jin Institute of Hydro graphic surveying and charting ,Tian Jin, 30061, China - - xin_xianhui@163.com
[e] He Nan Bureau of Surveying and Mapping, Zheng Zhou 450052, China- - zhangchunling06@sina.com
[f] State Key Library of Information Engineering in Survey, Mapping and Remote Sensing,Wuhan University, Wuhan, 430079,China - - guozhang@lmars.whu.edu.cn

Theme Sessions (ThS), ThS-4:

KEY WORDS: Classification, Detection, High-resolution, Image, SVM Method

ABSTRACT:

Automatic image categorization using low-level features is a challenging research topic in remote sensing application. In this paper, we formulate the image categorization problem as an image texture learning problem by viewing an image as a collection of regions, each obtained from image segmentation. Our approach performs an effective feature mapping through a chosen metric distance function. Thus the segmentation problem becomes solvable by a regular classification algorithm. Sparse SVM is adopted to dramatically reduce the regions that are needed to classify images. The selected regions by a sparse SVM approximate to the target concepts in the traditional diverse density framework. The proposed approach is a lot more efficient in computation and less sensitive to the class label uncertainty. Experimental results are included to demonstrate the effectiveness and robustness of the proposed method.

1. INTRODUCTION

Support Vector Machines (SVMs) are very attractive for the classification of remotely sensed data. The SVM approach seeks to find the optimal separating hyperplane between classes by focusing on the training cases that are placed at the edge of the class descriptors. These training cases are called support vectors. Training cases other than support vectors are discarded. This way, not only is an optimal hyperplane fitted, but also less training samples are effectively used; thus high classification accuracy is achieved with small training sets. This feature is very advantageous, especially for remote sensing datasets and Image Analysis, where samples tend to be less and less in number.

2. SUPPORT VECTOR MACHINES

The basic principles will be presented and then their implementation and application to Object Based Image Analysis will be evaluated. Let us consider a supervised binary classification problem. If the training data are represented by $\{x_i, y_i\}$, $i = 1, 2, ..., N$ and $y_i \in \{-1, +1\}$, where N is the number of training samples, $y_i = +1$ for class $\omega 1$ and $y_i = -1$ for class $\omega 2$. Suppose the two classes are linearly separable. This means that it is possible to find at least one hyperplane defined by a vector w with a bias w_0 , which can separate the classes without error:

$$f(x) = w \bullet x + w_0 \tag{1}$$

To find such a hyperplane, w and w_0 should be estimated in a way that $y_i(w \bullet x_i + w_0) \geq +1$ for $y_i = +1$ (class ω_1) and $y_i(w \bullet x_i + w_0) \leq -1$ for $y_i = -1$ (class ω_2). These two, can e combined to provide equation 2:

$$y_i(w \bullet x_i + w_0) - 1 \geq 0 \tag{2}$$

Many hyperplanes could be fitted to separate the two classes but there is only one optimal hyperplane that is expected to generalize better than other hyperplanes (Figure 1, Figure 2).

The goal is to search for the hyperplane that leaves the maximum margin between classes. To be able to find the optimal hyperplane, the support vectors must be defined. The support vectors lie on two hyperplanes which are parallel to the optimal and are given by:

$$w \bullet x_i + w_0 = \pm 1 \tag{3}$$

---

* Corresponding author. This is useful to know for communication with the appropriate person in cases with more than one author.

If a simple rescale of the hyperplane parameters w and w_0 takes place, the margin can be expressed as $\frac{2}{\|w\|}$ .The optimal hyperplane can be found by solving the following optimization problem:

Minimize: $\frac{1}{2}\|w\|^2$

Subject to: $y_i(w \bullet x_i + w_0) - 1 \geq 0 \quad i = 0,1,...,N$

Using a Lagrangian formulation, the above problem can be translated to:

$$\text{Maximize:} \quad \sum_{i=1}^{N} \lambda_i - \frac{1}{2}\sum_{i,j=1}^{N} \lambda_i \lambda_j y_i y_j (x_i \bullet x_j) \qquad (5)$$

$$\text{Subject to:} \quad \sum_{i=1}^{N} \lambda_i y_i = 0 \text{ and } \lambda_i \geq 0,\ i = 1,2,...,N$$

Where λ_i are the Lagrange multipliers.

Under this formulation, the optimal hyperplane discriminant function becomes:

$$f(x) = \sum_{i \in s} \lambda_i y_i (x_i x) + w_0 \qquad (6)$$

Where S is a subset of training samples that correspond to nonzero Lagrange multipliers. These training samples are called support vectors. In most cases, classes are not linearly separable, and the constrain of equation 2 cannot be satisfied. In order to handle such cases, a cost function can be formulated to combine

Maximization of margin and minimization of error criteria, using a set of variables called slack variables ξ (Figure 1,2). This cost function is defined as:

$$\text{Minimize:} \quad J(w, w_0, \xi) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N} \xi_i \qquad (7)$$

$$\text{Subject to:} \quad y_i(w \bullet x + w_0) \geq 1 - \xi_i \qquad (8)$$

To generalize the above method to non-linear discriminate functions, the Support Vector Machine maps the input vector x into a high-dimensional feature space and then constructs the optimal separating hyperplane in that space.

One would consider that mapping into a high dimensional feature space would add extra complexity to the problem. But, according to the Mercer's theorem, the inner product of the vectors in the mapping space, can be expressed as a function of the inner products of the corresponding vectors in the original space.

The inner product operation has an equivalent representation:
Maximize:

$$\sum_{i=1}^{N} \lambda_i - \frac{1}{2}\sum_{i,j=1}^{N} \lambda_i \lambda_j y_i y_j K(x_i \bullet x_j) \qquad (9)$$

Subject to:

$$\sum_{i=1}^{N} \lambda_i y_i = 0 \text{, and } \lambda_i \geq 0,\ i = 1,2,...,N$$

The resolution classifier becomes:

$$f(x) = \sum_{i \in S} \lambda_i y_i K(x_i x) + w_0 \qquad (10)$$

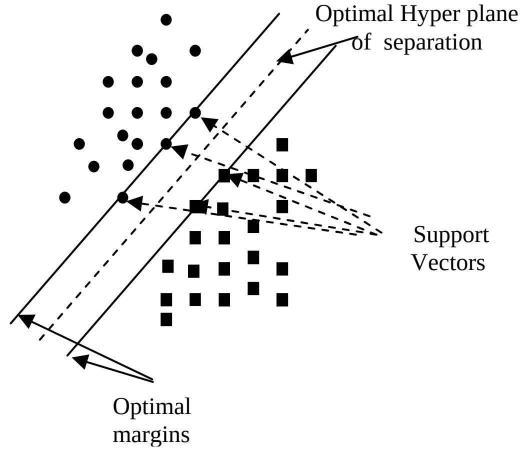

Figure 1: Linear separable classes

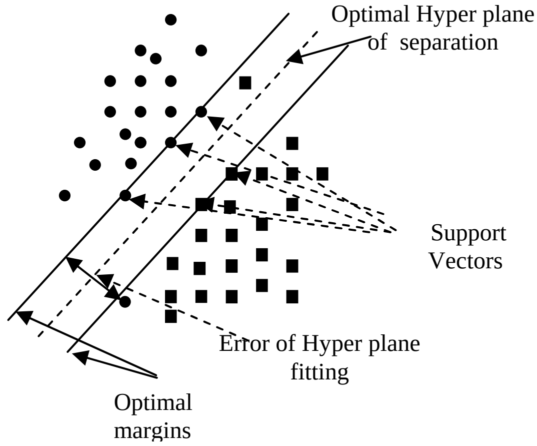

Figure 2: None linear separable classes

3. SVM MULTI CLASS CLASSIFICATION

The SVM method was designed to be applied only for two class problems. For applying SVM to multi-class classifications, two main approaches have been suggested. The basic idea is to reduce the multi-class to a set of binary problems so that the SVM approach can be used.

The first approach is called "one against all". In this approach, a set of binary classifiers is trained to be able to separate each class from all others. Then each data object is classified to the class for which the largest decision value was determined. This method trains N SVMs (where N is the number of classes) and there are N decision functions. Although it is a fast method, it suffers from errors caused by marginally imbalanced training sets.

Another approach was recently proposed, which is similar to the “one against all” method, but uses one optimization problem to obtain the N decision functions (equation 10). Reducing the classification to one optimization problem may require less support vectors than a multi-class classification based on many binary SVMs. The second approach is called “one against one”. In this, a series of classifiers is applied to each pair of classes, with the most commonly computed class kept for each object. Then a max-win operator is used to determine to which class the object will be finally assigned. The application of this method requires $N(N-1)/2$ machines to be applied. Even if this method is more computationally demanding than the “one against all” method, it has been shown that it can be more suitable for multi-class classification problems, thus it was selected for SVM object-based image classification.

4. CO-OCCURENCE FEATURES

The co-occurrence approach is based on the grey level spatial dependence. Co-occurrence matrix is computed by second order joint conditional probability density function $f(i,j\,|\,d,\theta)$. Each $f(i,j\,|\,d,\theta)$ is computed by counting all pairs of pixels separated by distance d having grey levels i and j, in the given direction θ. The angular displacement θ usually takes on the range of values from $\theta = 0, 45, 90, 135$ degrees. The co-occurrence matrix captures a significant amount of textural information. The diagonal values for a coarse texture are high while for a fine texture these diagonal values are scattered. To obtain rotation invariant features the co-occurrence matrices obtained from the different directions are accumulated. The three set of attributes used in our experiments are Energy, Inertia and Local Homogeneity.

$$E = \sum_i \sum_j [f(i,j\,|\,d,\theta)]^2 \tag{11}$$

$$I = \sum_i \sum_j [(i-j)^2 f(i,j\,|\,d,\theta)] \tag{12}$$

$$LH = \sum_i \sum_j \frac{f(i,j\,|\,d,\theta)}{1+(i+j)^2} \tag{13}$$

5. PROPOSED METHOD

The proposed Image Analysis worked in the following way: A training set of feature vectors was exported from manual segmentation and was used for training the SVM module. The SVM module is capable of using 4 types of kernels for training and classification:

Linear: $$K(x_i x_j) = x_i^T x_j \tag{14}$$

Polynomial: $$k(x_i x_j) = (\gamma \bullet x_i^T x_j + r)^d, \gamma > 0 \tag{15}$$

Radial Basis Function(RBF):

$$K(x_i x_j) = \exp(-\gamma \bullet \left\| x_i - x_j \right\|^2), \gamma > 0 \tag{16}$$

Sigmoid: $$K(x_i x_j) = \tanh(\gamma \bullet x_i^T x_j + r) \tag{17}$$

Where γ, r and d are kernel parameters.

All the above kernels follow Mercer’s theorem and can be used for mapping the feature space into a higher dimensional space to find an optimal separating hyper plane. In literature, there have been many comparison studies between the most common kernels. For pixel-based classification of remotely sensed data, it has been known that local kernels such as RBF can be very effective and accurate. Also, the linear kernel is a special case of the RBF kernel, with specific parameters. Based on the above, for the current study only RBF kernels were used. For the training of the SVM classifier, the error parameter C (7) and the kernel parameter γ had to be obtained. In order to find the optimal parameters for the RBF kernel function a cross-validation procedure was followed. First the training set was scaled to the range of $[-1,+1]$ to avoid features in greater numerical ranges dominating those in smaller ranges. Then, the training set was divided to many smaller sets of equal size. Sequentially each subset was tested using the classifier trained by the remaining subsets. This way each image object is predicted once during the above process. The overall accuracy of the cross-validation is the percentage of correctly classified image objects. After the cross-validation delivered the optimal parameters for the SVM classifier, the training set was used to train the SVM. Then the classifier was supplied with all image primitive so to derive the final classification

6. EXPERIMENTS AND CONCLUSION

For the evaluation of the developed approach, Ikonos images were used. Comparison with manual segmentation and got good results, we much improved efficiency of land resource investigation work when used it in interactive high resolution image segmentation. Overall, the SVM classification approach was found very promising for Image Analysis. It has been shown that it can produce comparable or even better results than the Nearest Neighbor for supervised classification. A very good feature of SVMs is that only a small training set is needed to provide very good results, because only the support vectors are of importance during training.

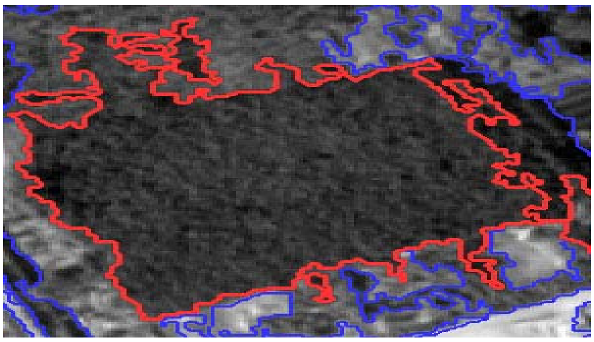

Figure 3 Regions of forest and region of farm

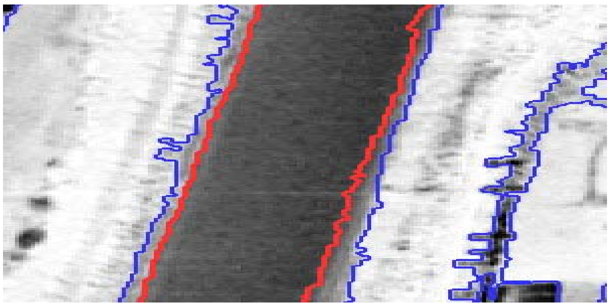

Figure 4 Regions different farms

ACKNOWLEDGEMENTS:

Thanks are due for the supports from:
Open research subject of Key Laboratory of Geo-informatics of State Bureau of Surveying and Mapping (Project No.: A1721).

China International Science and Technology Cooperation Project: High-Resolution Stereo Mapping Satellite: Field Geometric Calibration and Application (Project No.: 2006DFA71570).

China National 863 project: Intensive Image Matching and continuous digital surface reconstruction method based on high overlap aerial images (Grant No: 2006AA12Z134).

Commission of Science Technology and Industry for National Defense Project: Key Techniques of Data Processing for Mapping Satellite.

Natural Science Fund of P. R. China (No 40601084 and No. 40523005).

Open Research Fund Program of State Key Laboratory of Satellite Ocean Environment Dynamics (No. SOED0602).

Open Research Fund Program of the Geometrics and Applications Laboratory, Liaoning Technical University.

REFERENCES

K. P. Bennett, “Combining Support Vector and Mathematical Programming Methods for Classification,” B. Scholkopf, C.Burges and A. Smola, editors, Advances in Kernel Methods – Support Vector Machines, pp. 307–326, 1999.

J. Bi, K. P. Bennett, M. Embrechts, C. Breneman and M.Song, “Dimensionality Reduction via Sparse Support Vector Machines,” Journal of Machine Learning Research, 3:1229–1243, 2003.

O. Chapelle, P. Haffner, and V. N. Vapnik, “Support Vector Machines for Histogram-Based Image Classification,” IEEE Transactions on Neural Networks, 10(5):1055–1064, 1999.

Y. Chen and J. Z. Wang, “Image Categorization by Learning and Reasoning with Regions,” Journal of Machine Learning Research, 5:913–939, 2004.